

PREFACE

UNITED STATES NAVAL OBSERVATORY

CIRCULAR NO. 121

U. S. Naval Observatory, Washington, D. C. 20390

June 28, 1968

Beginning with the Air Almanac 1945, camera copy of exceptional quality was produced by a specially designed card-operated typewriter ("The Printing of Mathematical Tables" by W. J. Eckert and R. F. Haupt; Mathematical Tables and Other Aids to Computation, Vol. II, No. 17, Jan. 1947).

In 1962 the Nautical Almanac Office was the first agency in the Department of Defense to bridge the gap between electronic data processing and automatic photo-composition for offset printing ("Printing of Astronomical and Sight Reduction Tables" by R. L. Duncombe and R. F. Haupt; Navigation, Vol. 12, No. 2, Jan. 1963). The Nautical Almanac Office has since then produced camera copy for the Office Linofilm and Linotron equipment over 4000 pages of astronomical and navigational tables. The technical details of the Linotron system were made available at the G.P.O. through the cooperation of Mr. John Boyle, Special Assistant to the Director.

A LINOTRON SYSTEM MANUAL FOR THE  
PHOTO - COMPOSITION OF  
ASTRONOMICAL AND MATHEMATICAL TABLES

As a continuation of this effort, the following instruction manual by D. K. Scott provides basic information and programming instructions for control of a Linotron electronic composing system. Inquiries regarding this manual and the availability of the programming subroutines described should be directed to one of the undersigned.

by

D. K. Scott

R. F. Haupt  
Assistant Director  
Nautical Almanac Office

R. L. Duncombe  
Director  
Nautical Almanac Office



THE UNIVERSITY OF CHICAGO

PHOTOGRAPHY

1950

ASTRONOMICAL OBSERVATORY

A PHOTOGRAPH OF THE

PHOTOGRAPHY

ASTRONOMICAL OBSERVATORY

1950



## PREFACE

The Nautical Almanac Office of the U. S. Naval Observatory first introduced punched-card electric accounting machine equipment for scientific computing and the production of printer's copy for its publications during the year 1940. Printer's copy for the Air Almanac 1941 was produced by this equipment. Camera copy for offset printing was automatically produced on a 401 tabulator for the Air Almanac 1942 ("Air Almanacs" by W. J. Eckert; Sky and Telescope, Vol. IV, No. 1, Nov. 1944).

Beginning with the Air Almanac 1946, camera copy of exceptional quality was produced by a specially designed card-operated typewriter ("The Printing of Mathematical Tables" by W. J. Eckert and R. F. Haupt; Mathematical Tables and Other Aids to Computation, Vol. II, No. 17, Jan. 1947).

In 1962 the Nautical Almanac Office was the first agency in the Department of Defense to bridge the gap between electronic data processing and automatic photo-composition for offset printing ("Printing of Astronomical and Sight Reduction Tables" by R. L. Duncombe and R. F. Haupt; Navigation, Vol. 12, No. 2, Summer 1965). Since that time the Nautical Almanac Office has composed on the Government Printing Office Linofilm and Linotron equipment over 4000 pages of astronomical and navigational tables. The technical details on programming for the Linotron system were made available at the G.P.O. through the cooperation of Mr. John Boyle, Special Assistant to the Production Manager for Electronic Printing.

As a continuation of this effort, the following instruction manual by D. K. Scott provides basic information and programming instructions for control of a Linotron electronic composing system. Inquiries regarding this manual and the availability of the programming subroutines described should be directed to one of the undersigned.

R. F. Haupt  
Assistant Director  
Nautical Almanac Office

R. L. Duncombe  
Director  
Nautical Almanac Office







# A LINOTRON SYSTEM MANUAL

FOR

## PHOTO-COMPOSITION OF ASTRONOMICAL AND MATHEMATICAL TABLES

### INTRODUCTION

The Linotron is a new, rapid electronic phototypesetting machine which was installed at the Government Printing Office late in the summer of 1967. It was manufactured by the Mergenthaler Linotype Company and Columbia Broadcasting System. It operates off-line from a computer using magnetic tape as input. The Linotron obtains the characters to be composed from one of four character grids, each of which contains 255 characters. Any character can be generated in random order and composed in eight different type sizes. Speed of composition ranges between 300 to 1000 characters per second. Vertical and horizontal rules can also be drawn. The maximum image size is approximately  $8 \times 10\frac{1}{2}$  inches. The details of how characters are actually transferred from the grids to the photocathodes are outside the scope of this report.

The aim of the Naval Observatory Linotron system is to prepare magnetic tapes for Linotron which will operate the machine directly. The Government Printing Office and the manufacturers of the machine have a "Master Typography Program" designed to convert magnetic tape containing both data and "editing" codes into the appropriate text and control characters to operate Linotron. Much of this MTP is necessarily oriented to processing textual material. Since our data are nearly all tabular and numerical with headings and rules, it usually will result that the direct approach is simpler as well as saving the extra program run needed by the MTP.

It will be necessary to use printer's measures which will be explained briefly. The basic unit is the "point". One point is equal to .013837 inch, or very nearly  $1/72$  inch. Twelve points are called a "pica" (approximately  $1/6$  inch) and this unit is usually used to indicate the dimensions of pages or columns. Type sizes are expressed in points. The Linotron system can set type in 5-, 6-, 7-, 8-, 10-, 12-, 14-, and 18-point sizes. An "em" is a square the size of the type being set; an "en" is a rectangle whose short side is half the width of an em. An "em quad" is a blank the size of an em.

In photocomposing the term "unit" is also used. It has two meanings. The first is used to describe the width of characters in terms of an em quad, which is divided into 18 units (e.g. a capital M may have a width of 15 units; numerals a width of 8, 9, or 10 units; a comma a width of 4 units). Characters have a minimum width of 4 units, a maximum width of 18 units.

The second meaning of unit is the smallest unit of resolution of the system, defined as  $1/18$  of a point (.000769 inch). To convert a character width given in relative units to absolute units, multiply by the type size. Thus a 9 unit width set in 5-point will have a width of 45 units; if set in



14-point, 126 units. Coordinates are expressed in units rather than in points, picas, or inches. For many purposes the following table will be useful:

|         |              |
|---------|--------------|
| 1 point | = 18 units   |
| 1 pica  | = 216 units  |
| 1 inch  | = 1300 units |

The Linotron system will allow a frame as large as 48 x 63 picas to be composed. Rectangular coordinates are used to locate points. The origin is the upper left-hand corner of the frame; y increases to the bottom of the page, x increases to the right. There are no negative coordinates. When locating a point its y-coordinate is given first, followed by its x-coordinate. There is also a provision for indicating values of delta-y or delta-x in the positive direction. The effect of a negative delta-x or delta-y can be obtained by using a number greater than 20000, but we have not used this feature. When negative delta values are needed it will usually be simpler to perform the necessary arithmetic and to assign new y- and x-coordinates, since both will require the same number of characters.

The characters used by Linotron fall into two classes. The first is used to indicate where on the grid a particular character is to be found. The arrangement used is to divide each grid into three zones containing 85 characters. If the grid is a "standard" one, each zone will contain the characters for a complete alphabet (caps and lower case), ten numerals, and a selection of punctuation characters. There is also a designation of shift and unshift positions. If the grid is in standard arrangement, then the numerals and the capital and lower case letters are designated by the BCD numeral or letter that they represent. Letters are used in both shift and unshift mode; numerals are designated in the unshift mode. Numerals in the shift mode are used for small capitals or punctuation. Beside the letters and numerals, eleven other BCD special characters are also used in the shift and unshift mode.

With this selection of 47 BCD characters in two modes, 94 character positions could be selected. Since there are only 85 characters in each zone, nine shift and unshift positions are restricted for certain of the numerical and special characters in each zone. The restricted positions are different for each of the three zones. The exact arrangement for each of the three zones will be given in the explanation of the LINGRID and LINACC subroutines.

As previously mentioned, the BCD representations of letters and numerals will usually be used to represent the same letters and numerals on the grid. To some extent this will be true for special characters but there will be some exceptions. Thus on one grid the \$ symbol is assigned the designation unshift %, although the BCD character \$ is also a valid character code.

The Naval Observatory grid contains only two alphabets, the remainder



of the characters being special characters needed for the Air Almanac, H.O. Pub. No. 229, H.O. Pub. No. 249, and a selection of Greek letters and mathematical and astronomical symbols. At least 66 of these will be designated by BCD letters and numerals.

Each of the three zones will be selected by the Zone Recognition code (#, 3-8, 7B) followed by the numeral 1, 2, or 3. The unshift code is the word separator (0-5-8, 6D); the shift code is the record mark (0-2-8, EO). Whenever a particular zone or shift code has been designated, it remains in effect until it is replaced by another code. The characters indicating text are then laid out consecutively with zone and shift codes introduced only when a change takes place. This means that alphabetic and numerical text can be read directly from a printout of the tape.

The eleven BCD special characters that have been used to indicate grid position are:

| EBCDIC | Card Punch | Hexadecimal | EBCDIC | Card Punch | Hexadecimal |
|--------|------------|-------------|--------|------------|-------------|
| .      | 12-3-8     | 4B          | ,      | 0-3-8      | 6B          |
| <      | 12-4-8     | 4C          | %      | 0-4-8      | 6C          |
| &      | 12         | 50          | >      | 0-6-8      | 6E          |
| \$     | 11-3-8     | 5B          | ?      | 0-7-8      | 6F          |
| -      | 11         | 60          |        | 12-0       | CO          |
| /      | 0-1        | 61          |        |            |             |

The second class of characters are the control characters. In order to have a sufficient number, two special characters have been designated as precedence codes. These are introduced as needed; once entered they remain in effect until replaced by the opposite code. The code for precedence A is the logical not (11-7-8, 5F); when in effect the control codes which follow control text composition or the text composition itself. The code for precedence B is the colon (2-8, 7A); when in effect the control codes which follow are functions to be performed when not composing text, such as frame recognition, grid change, stop, etc. (See LINOTRON FUNCTION CODE REPERTOIRE, page 39.)

A description of each code and the associated characters which must follow will be described next. It should be noted that the same character can have a different control function depending on which precedence is in effect.

### Description of Function Codes.

The following functions are performed in Precedence A:

1. Shift  
0-2-8 EO      The character(s) which follow are set in the shift mode.
2. Unshift  
0-5-8 6D      The character(s) which follow are set in the unshift mode.

Codes 1 and 2 are also valid in precedence B.



3. Y-Origin      The y-coordinate is to be set to a new value.  
 = 6-8 7E      The code is to be followed by five BCD numerals  
                  which indicate the new y-coordinate. All lead-  
                  ing zeros must be included.
4. X-Origin      The x-coordinate is to be set to a new value.  
 " 7-8 7F      It is to be followed by five BCD numerals that  
                  indicate the new x-coordinate.

Codes 3 and 4 are used each time that the location on the frame is to be respecified at the beginning of a line or the starting point of a rule. The y-origin is always specified first, followed by the x-origin. Once specified the character symbols which follow are set in order from left to right. It should be noted that the y-coordinate refers to the "z-line" for the character which lines with the base of such characters as lower case p, q, and y. (See the type specimens, pages 41 and 43; a rule has been printed at the z-line). Numerals and most capital and lower case letters have considerable white space between the z-line and the base of the characters. The x-coordinate refers to the left edge of the type character.

5. Point Size Recognition      This code is used to indicate in  
 @ 4-8 7C      what size type the characters which follow are  
                  to be set. One of the following characters must  
                  follow the recognition code:

|   |          |       |
|---|----------|-------|
| 5 | 5-point  |       |
| 6 | 6-point  |       |
| 7 | 7-point  |       |
| 8 | 8-point  |       |
| 0 | 10-point |       |
| @ | 12-point | (4-8) |
| = | 14-point | (6-8) |
| S | 18-point |       |

Note that in each case the point size is indicated in a five-level binary code based on the BCD (6-level) representation of each character.

6. Non-Print      When this code is in effect the characters which  
 ( 12-5-8 4D      follow are skipped over by the composing unit  
                  without printing although the distance moved is  
                  the same as if the character had been printed.  
                  One use for this code is in zero suppression or  
                  the suppression of leading digits.
7. Print      This character is necessary to cause characters  
 \* 11-4-8 5C      to be actually printed. It should follow the Non-  
                  Print code when printing is to be resumed.
8. Delta-Y      The y-coordinate is to be incremented while the  
 ; 11-6-8 5E      x-coordinate remains unchanged. It is to be  
                  followed by five BCD numerals giving delta-y in  
                  units.



9. Delta-X      The x-coordinate is to be incremented while the  
 11-0    DO      y-coordinate remains fixed. It is to be followed  
                  by five BCD numerals indicating delta-x in units.  
                  This is used to give the spacing between words,  
                  columns, etc.
10. Zone Recognition      This code is used to specify which of the  
 #      3-8    7B      three main divisions on the grid is to be used.  
                  It must be followed by one of the three numbers  
                  1, 2, or 3.
11. Rule Recognition      This code is used to indicate that rules are  
 )      11-5-8    5D      to be drawn. It should be followed by one of the  
                  following codes to indicate the width of the rules:
- |   |                       |
|---|-----------------------|
| 1 | .003 inch width rules |
| 2 | .007 inch width rules |
| 4 | .015 inch width rules |

Following these two characters it is necessary to have a y-origin, x-origin, delta-x to specify a horizontal rule, or a y-origin, x-origin, delta-y to specify a vertical rule.

Note that 20 characters are used to specify each rule. The Rule Recognition and width should be repeated for each rule set.

The x-origin and delta-x should end in either zero or five when drawing horizontal rules. Similarly y-origin and delta-y should end in zero or five when drawing vertical rules. The point size should have been set to 5-point before drawing rules.

The following functions are performed in Precedence B:

1. Job Recognition      This code must appear as the first item at  
 @      4-8    7C      the beginning of each frame. It is to be followed  
                  by a six-digit BCD numeral, which ordinarily will  
                  be the G.P.O. job jacket number assigned to the  
                  particular project.
2. Frame Recognition      This code must appear following the Job Recog-  
 =      6-8    7E      nition code and the job number. It is to be followed  
                  by a four-digit BCD numeral indicating the number of  
                  the frame. The Job and Frame Recognition codes with  
                  their accompanying numerals are used by the Linotron  
                  Photocomposer under console control to perform such  
                  functions as searching for a starting frame, stop-  
                  ping at a particular point, and so forth.
3. Grid Recognition      This code is used to indicate a grid change.  
 #      3-8    7B      The system may use up to four grids for any given  
                  job. The Grid Recognition code must be followed



by a BCD numeral to indicate which turret the desired grid has been placed. The codes are as follows:

|   |          |
|---|----------|
| 1 | Turret 1 |
| 2 | Turret 2 |
| 4 | Turret 3 |
| 8 | Turret 4 |

On frames requiring more than one grid, all copy from a grid should be composed at once, since all grid changes are mechanical and require a large fraction of a second, during which several hundred characters could have been set.

4. Page Orient  
; 11-6-8 5E

Type may be composed in either the Normal or Turn Page mode. In the Turn Page mode the x- and y-coordinates are interchanged. The Page Orient code must be followed by a BCD numeral 2 to indicate Normal mode or BCD numeral 1 to indicate Turn Page mode.

When a whole page is to be set in Turn Page mode, no special problem arises except that the new y-maximum is 48 picas (10368 units) and the new x-maximum is 63 picas (13608 units). When setting a Turn Page section on a page which has been set in Normal mode, it should be noted that Turn Page involves a clockwise rotation so that the new y-coordinate is equal to the Normal mode x-coordinate. For setting a type line note that some adjustment may be necessary for the thickness of the characters. The new x-coordinate is equal to (13608 minus Normal y-coordinate).

When setting pages using both orientations, the Turn Page and Normal copy should be composed so that all copy in one orientation is composed together so as to minimize the time lost when shifting from one to the other.

5. Aspect Ratio  
" 7-8 7F

The Aspect Ratio code may be used to condense the width of the type face and obtain a pseudo condensed type. This code must be followed by one of three BCD numerals to produce the condensation shown:

|   |                      |
|---|----------------------|
| 1 | 100% of normal width |
| 2 | 83% of normal width  |
| 4 | 63% of normal width  |

(Examples of three sets of figures printed in the three different Aspect Ratios appear on the type specimens, pages 41 and 43.)



6. Special Face      The Special Face code is used to change the scan ratio so as to produce "synthetic bold" in which the lines of the character are made darker, and "synthetic italic" in which the character is artificially tilted. The Special Face code must be followed by one of the following:
- |   |        |    |       |                                       |
|---|--------|----|-------|---------------------------------------|
| ( | 12-5-8 | 4D | 1     | Normal face                           |
|   |        |    | 4     | Synthetic italic                      |
|   |        |    | 8     | Synthetic bold                        |
|   |        |    | (4-8) | @ Synthetic bold and italic combined. |
- (Examples of synthetic bold and italic figures appear on the type specimens, pages 41 and 43.)
7. Film Advance      The Film Advance code is used at the end of each frame to advance the photographic medium on which the job is being printed. The amount of the advance is indicated by one of the four BCD numerals to indicate the advance shown:
- |   |        |    |   |                                               |
|---|--------|----|---|-----------------------------------------------|
| * | 11-4-8 | 5C | 1 | 6-inch film advance                           |
|   |        |    | 2 | 7-inch film advance                           |
|   |        |    | 4 | 8-inch film advance                           |
|   |        |    | 8 | 8 $\frac{1}{2}$ -inch film advance (maximum). |
8. Focus              The Focus code is used to cause the photocomposer to check the focus on a point near the middle of the frame as part of the start-up procedure. The point focussed on has the coordinates  $y = 07170$ ;  $x = 06830$ .
- F 12-6      06
9. Stop                The Stop code causes the photocomposer to stop.
- 11-0      DO
10. Overlay            This code is used to indicate that a special overlay is to be projected. (No further information is available about this special feature).
- , 0-3-8      6B

### Tape Record.

Each tape record for the Linotron system is to be 2048 characters long, written in 7-track, odd-parity, 556 cpi. Each frame must start a new record, and begin with Job and Frame Recognition codes and the associated job and frame numbers. The other coding required to start a frame will be described in detail in the description of the LINFRAME subroutine. After these "startup" constants have been taken care of, then the "text" for each frame can be set in almost any order. The scheme used for each job will depend largely on the arrangement of the input records, the necessity for corrections and additional computations, and so forth. Besides this, the main consideration is that all text to be composed from any one grid should be composed at one time, before shifting to another grid, in order to minimize the time lost from grid changes. Since it is not necessary to set



characters contiguously, there should be no difficulty in following this rule. At the end of each frame it is necessary to assemble a Film Advance code and then to pad the remainder of the 2048 character record with blanks. A blank may be used anywhere for padding purposes; the Linotron photocomposer just skips to the next non-blank character. It happens that in the composing "system" to be described few blanks are set. This will have the advantage of getting the maximum number of useful characters on each tape record.

All of the programs to be described are for use on the computer at the Naval Observatory, an IBM System/360, Model 40. It is assumed that the reader is familiar with the System/360 Assembler Language, program layout including the standard methods used to call subroutines and to perform I/O operations, hexadecimal notation, and similar topics.

Six subroutines that have been developed will now be described. The name of each of them begins with the letters "LIN-". The use of these subroutines is in line with the basic System/360 philosophy of breaking programs into short segments and to make elaborate use of addresses to connect the segments. Most of the large blocks of storage that will be required are handled by the subroutines, so that the "main" program required for each job will usually be quite short. These subroutines have not been used enough to be thoroughly debugged and some desirable features may be lacking. Users are urged to find out for themselves how each subroutine works so that they can make changes as required. It is also expected that additions will be made to the Linotron subroutine "library". All of these subroutines have been assembled and are available in object decks which can be loaded with main programs. The identification for each object deck has been indicated.

#### LINWRITE Subroutine:

The purpose of this subroutine is to perform all of the operations necessary to write the 2048 character, 7-track binary tape records, including the necessary open and close I/O operations, to perform the control operation, and to provide the necessary core storage in which the 2048 character record is assembled. The DCB, which is for BSAM, is included and contains the following operands:

|        |              |         |                |
|--------|--------------|---------|----------------|
| DSORG  | PS           | TRTCH   | T (Odd Parity) |
| MACRF  | (WC,RC)      | RECFM   | F              |
| DDNAME | LINOTAPE     | BUFNO   | 2              |
| DEVN   | TA           | LRECL   | 2048           |
| DEN    | 1 (556 cpi.) | BLKSIZE | 2048           |

Four different functions can be performed by this subroutine. The one to be performed is to be indicated by an option byte, the address of which is always the first parameter. The first call to the subroutine will cause an OPEN Macro (INOUT) to be executed.

#### Option 1.

This option is used to space the tape a given number of records to find the starting point for a continuation run. Two parameters are



required:

1. Address of option byte. Bit structure: 1000 0000 80
2. Address of fullword with the number of records to be spaced.

If the tape does not contain the number of records specified in parameter 2, the result will probably be an I/O error when blank records are read.

#### Option 2.

This option is used to transfer any number of characters to the 2048-byte block from which the record is written onto tape. When transferring characters, if the 2048-byte block becomes full, a tape record is automatically written. When the character string is exhausted, any excess over 2048 is left in place to be written ahead of the next character string, transferred using option 2 or 3. Two parameters are required:

1. Address of option byte. Bit structure: 0000 0000 00
2. Address of fullword followed by a halfword. The fullword must contain the address of the character string, the halfword the number of bytes in the string. (The maximum halfword, 32767, is the limit on the number of bytes that can be transferred.) The LINCOMP and LINHEADS subroutines both return their results in the form that can be used by this subroutine.

#### Option 3.

This option is the same as option 2, but in addition, when the last of the character string has been transferred to the 2048-byte tape area, a film advance code is added and the remainder of the 2048 bytes is padded with blanks and the block written to tape. The next call to the LINWRITE subroutine will start a new 2048-byte block. Two parameters are required:

1. Address of option byte. Bit structure: 0000 0001 01
2. Same as option 2.

Before using this option it is necessary that the proper value for the film advance code be placed in the constant area in the subroutine. This can be done by the LINFRAMX subroutine, which should have been called before using this option.

#### Option 4.

This option is used to write terminal Linotron records (two film advances and a stop code), pad the remainder of the 2048 bytes with blanks, write tape, and close I/O. One parameter is required:

1. Address of option byte. Bit structure: 1111 1111 FF

After closing I/O, the following message will be printed on the console typewriter: "OUTPUT TAPE CLOSED; xxxx blocks written." A running count is kept of all records as they are written on the tape; this count will



include the number spaced up using option 1, so that the total should equal the number of records on the tape.

If any option byte other than the four shown above is used, the subroutine will return to the main program after having done nothing. No error will be indicated.

It will be necessary to have a DD card in the following configuration with the program:

```
//GO.LINOTAPE DD UNIT=180,DISP=(OLD),LABEL=(,BLP),VOLUME=SER=xxx.
```

The subroutine uses 2963 bytes. Its four-digit identification is LNWR.

#### LINFRAME (LINFRAMX) Subroutine.

This subroutine performs two functions. It allows the programmer (1) to insert the parameters that are necessary to start a frame (Linotron startup constants) into the subroutine work area, or (2) to start each new frame while running a job by transferring the assembled Linotron startup constants to the LINWRITE subroutine. The subroutine has two entry points.

Entry LINFRAME (no parameter) will perform function (2) above. It should be called as the first operation at the start of programming for each frame. Each call to the subroutine will cause the frame number to be incremented by one after the call to LINWRITE has been completed.

Entry LINFRAMX will require one parameter, the address of a 41-byte, 15-field character string, the items for which will be described below.

As previously mentioned, it is necessary that each frame begin with a Precedence B code, followed by a Job Recognition code, a six-digit job number, a Frame Recognition Code, and a four-digit frame number. After this, the format to be used is not absolutely fixed by the requirements of the Linotron machine, but for convenience we have determined a format for an 84-byte field which will satisfy the minimum startup constants that are required. Some of these characters are fixed; the variable characters will have been provided from the entry LINFRAMX.

The general format will be described first. Characters which are to be inserted will be underlined. This format provides for printing the job and frame number somewhere on the frame. Ordinarily these will be printed in the lower left hand corner of the frame and in fairly large type (say 12-point). The format used for printing the job number includes a hyphen or en dash between the first and last group of three figures (e.g. 244-122). The grid, type size, zone, and shift codes to be inserted are to specify the figures to be printed at this time. Usually, the grid number should also refer to the first section of type to be composed.



## GENERAL FORMAT

| Function                    | Character                  | Hexa-<br>decimal | Remarks                                                                                                                                                                      |
|-----------------------------|----------------------------|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Precedence B             | :                          | 7A               |                                                                                                                                                                              |
| 2. Job Recognition          | @                          | 7C               |                                                                                                                                                                              |
| 3. Job Number               | <u>XXXXXX</u>              |                  | Six BCD numerals                                                                                                                                                             |
| 4. Frame Recognition        | =                          | 7E               |                                                                                                                                                                              |
| 5. Frame Number             | <u>XXXX</u>                |                  | Four BCD numerals                                                                                                                                                            |
| 6. Special Face Recog.      | (                          | 4D               |                                                                                                                                                                              |
| 7. Face Code                | <u>1,4,8</u> or @          |                  | See the description for<br>Special Face                                                                                                                                      |
| 8. Aspect Ratio Recog.      | "                          | 7F               |                                                                                                                                                                              |
| 9. Aspect Code              | <u>1,2,4</u>               |                  | 1=100%; 2=83%; 4=63%                                                                                                                                                         |
| 10. Page Orient Recog.      | ;                          | 5E               |                                                                                                                                                                              |
| 11. Page Orient Code        | <u>1</u> or <u>2</u>       |                  | 1=Turnpage; 2=Normal                                                                                                                                                         |
| 12. Precedence A            | -                          | 5F               |                                                                                                                                                                              |
| 13. Y-Origin                | =                          | 7E               |                                                                                                                                                                              |
| 14. Focus Y                 | 07170                      |                  | Coordinates near center of<br>frame on which focus check<br>is made.                                                                                                         |
| 15. X-Origin                | "                          | 7F               |                                                                                                                                                                              |
| 16. Focus X                 | 06830                      |                  |                                                                                                                                                                              |
| 17. Precedence B            | :                          | 7A               |                                                                                                                                                                              |
| 18. Focus Recognition       | F                          | C6               |                                                                                                                                                                              |
| 19. Grid Recognition        | #                          | 7B               |                                                                                                                                                                              |
| 20. Turret Number           | <u>1,2,4</u> , or <u>8</u> |                  |                                                                                                                                                                              |
| 21. Precedence A            | -                          | 5F               |                                                                                                                                                                              |
| 22. Zone Recognition        | #                          | 7B               |                                                                                                                                                                              |
| 23. Zone                    | <u>1,2</u> , or <u>3</u>   |                  | Zone and shift code must be<br>for a set of numerals on the<br>grid selected.                                                                                                |
| 24. Shift Code              | <u>0-5-8,0-2-8</u>         | 6D E0            |                                                                                                                                                                              |
| 25. Point Size Recog.       | @                          | 7C               |                                                                                                                                                                              |
| 26. Size Code               | <u>5,6,7,8,0,@,-,S</u>     |                  | Twelve-point,@, is recom-<br>mended.                                                                                                                                         |
| 27. Print On                | *                          | 5C               |                                                                                                                                                                              |
| 28. Y-Origin                | =                          | 7E               |                                                                                                                                                                              |
| 29. Y-Coordinate Job No.    | <u>XXXXX</u>               |                  |                                                                                                                                                                              |
| 30. X-Origin                | "                          | 7F               |                                                                                                                                                                              |
| 31. X-Coordinate Job No.    | <u>XXXXX</u>               |                  |                                                                                                                                                                              |
| 32. Job Number              | <u>XXX-XXX</u>             |                  | The code for the numerals will<br>always be numerals, but the<br>code for the hyphen will vary<br>from grid to grid and must be<br>specified correctly by the<br>programmer. |
| 33. Y-Origin                | =                          | 7E               |                                                                                                                                                                              |
| 34. Y-Coordinate, Frame No. | <u>XXXXX</u>               |                  |                                                                                                                                                                              |
| 35. X-Origin                | "                          | 7F               |                                                                                                                                                                              |
| 36. Y-Coordinate, Frame No. | <u>XXXXX</u>               |                  |                                                                                                                                                                              |
| 37. Frame Number            | <u>XXXX</u>                |                  |                                                                                                                                                                              |

These codes require 78 bytes; space for an additional 6 bytes is left  
if the symbol used for the hyphen or en dash between the two numerical groups  
in the job number should be in a different zone or shift than the numerals.



The characters to appear in each of the fifteen fields are listed below. The address of the first byte of the 41-character field is the parameter for LINFRAMX.

| Field | Position | Description and Remarks                                    | Position in Startup Constant Field |
|-------|----------|------------------------------------------------------------|------------------------------------|
| 1.    | 1-6      | *Job Number 6 numerals                                     | 3; 32                              |
| 2.    | 7-10     | **Frame Number 4 numerals                                  | 5; 37                              |
| 3.    | 11       | ***Special Face Code                                       | 7                                  |
| 4.    | 12       | ***Aspect Ratio Code                                       | 9                                  |
| 5.    | 13       | ***Page Orient Code                                        | 11                                 |
| 6.    | 14       | Turret Number                                              | 20                                 |
| 7.    | 15       | Zone Number                                                | 23                                 |
| 8.    | 16       | Shift or Unshift                                           | 24                                 |
| 9.    | 17       | Type Size                                                  | 26                                 |
| 10.   | 18-22    | +Y <sub>o</sub> for Job Number Printout                    | 29                                 |
| 11.   | 23-27    | X <sub>o</sub> for Job Number Printout                     | 31                                 |
| 12.   | 28-32    | Y <sub>o</sub> for Frame Number Printout                   | 34                                 |
| 13.   | 33-37    | X <sub>o</sub> for Frame Number Printout                   | 36                                 |
| 14.   | 38-40    | Zone, Shift/Unshift and Char. used for hyphen on grid used | 32                                 |
| 15.   | 41       | ++Film Advance Code                                        |                                    |

Each item in the list above must be provided. There are no default options. It should be noted that all of the parameters to be furnished in the list are the actual Linotron codes.

- \* The job number will usually be the job-jacket number that has been assigned to the particular job by the G.P.O.
- \*\* The frame number should be started at 0001 for an initial run. When starting a continuation run this parameter should be changed to continue from the last frame already on the tape.
- \*\*\*It generally will be desirable to use the codes 112 for these three fields, items 7,9, and 11, in the general format list above, indicating normal face, normal width, and normal page orientation. The programmer should not forget to start his type composition with a Precedence B code, the codes for special face, etc., followed by a Precedence A code if any of these items should be different than what is wanted for the regular job.
- + Provision has been made for separate y- and x-origins for the printout of both job and frame numbers. This will allow the two items to be set on the same line, or one item above the other.
- ++ The film advance code is not one of the startup constants. Like a startup constant it varies from job to job and this allows the programmer to specify it at the same time as the other parameters. This subroutine inserts the film advance code into a constant area in the LINWRITE subroutine. Ordinarily one should use the smallest film advance that will pull his film out of the way.

This subroutine requires 445 bytes. Its identification on the object deck is LFRM.



LINGRID Subroutine.

As previously explained, 47 BCD letters, numerals, and special characters are used to designate the 255 characters on each grid. In order to differentiate between these characters, the grid is divided into three zones, and within each zone the character may occur in either unshift or shift mode. Thus there are three characters associated with each of the 255 positions on the grid. The designation for each character position on any grid is the same, although the characters themselves may differ. The one thing that will differ is the relative width associated with each character. For use with composition subroutines (such as LINCOMP subroutine), the scheme used is to represent each of the 255 characters on the grid by a word (byte 0 = zone code, byte 1 = shift or unshift, byte 2 = Linotron character, byte 3 = relative width in units as a binary number). Since the first three bytes in each word will be the same for all grids, the only information to be transmitted to the subroutine will be the relative widths for all characters on a particular grid. The grid positions have been numbered from 001 to 255 and arranged as shown in Table 1.

Table 1

| No. | Zone | U/S | Char. | No. | Zone | U/S | Char. | No. | Zone | U/S | Char. |
|-----|------|-----|-------|-----|------|-----|-------|-----|------|-----|-------|
| 001 | 1    | U   | A     | 086 | 2    | U   | A     | 171 | 3    | U   | A     |
| 002 | 1    | U   | B     | 087 | 2    | U   | B     | 172 | 3    | U   | B     |
| 003 | 1    | U   | C     | 088 | 2    | U   | C     | 173 | 3    | U   | C     |
| 004 | 1    | U   | D     | 089 | 2    | U   | D     | 174 | 3    | U   | D     |
| 005 | 1    | U   | E     | 090 | 2    | U   | E     | 175 | 3    | U   | E     |
| 006 | 1    | U   | F     | 091 | 2    | U   | F     | 176 | 3    | U   | F     |
| 007 | 1    | U   | G     | 092 | 2    | U   | G     | 177 | 3    | U   | G     |
| 008 | 1    | U   | H     | 093 | 2    | U   | H     | 178 | 3    | U   | H     |
| 009 | 1    | U   | I     | 094 | 2    | U   | I     | 179 | 3    | U   | I     |
| 010 | 1    | U   | J     | 095 | 2    | U   | J     | 180 | 3    | U   | J     |
| 011 | 1    | U   | K     | 096 | 2    | U   | K     | 181 | 3    | U   | K     |
| 012 | 1    | U   | L     | 097 | 2    | U   | L     | 182 | 3    | U   | L     |
| 013 | 1    | U   | M     | 098 | 2    | U   | M     | 183 | 3    | U   | M     |
| 014 | 1    | U   | N     | 099 | 2    | U   | N     | 184 | 3    | U   | N     |
| 015 | 1    | U   | O     | 100 | 2    | U   | O     | 185 | 3    | U   | O     |
| 016 | 1    | U   | P     | 101 | 2    | U   | P     | 186 | 3    | U   | P     |
| 017 | 1    | U   | Q     | 102 | 2    | U   | Q     | 187 | 3    | U   | Q     |
| 018 | 1    | U   | R     | 103 | 2    | U   | R     | 188 | 3    | U   | R     |
| 019 | 1    | U   | S     | 104 | 2    | U   | S     | 189 | 3    | U   | S     |
| 020 | 1    | U   | T     | 105 | 2    | U   | T     | 190 | 3    | U   | T     |
| 021 | 1    | U   | U     | 106 | 2    | U   | U     | 191 | 3    | U   | U     |
| 022 | 1    | U   | V     | 107 | 2    | U   | V     | 192 | 3    | U   | V     |
| 023 | 1    | U   | W     | 108 | 2    | U   | W     | 193 | 3    | U   | W     |
| 024 | 1    | U   | X     | 109 | 2    | U   | X     | 194 | 3    | U   | X     |
| 025 | 1    | U   | Y     | 110 | 2    | U   | Y     | 195 | 3    | U   | Y     |
| 026 | 1    | U   | Z     | 111 | 2    | U   | Z     | 196 | 3    | U   | Z     |
| 027 | 1    | S   | A     | 112 | 2    | S   | A     | 197 | 3    | S   | A     |
| 028 | 1    | S   | B     | 113 | 2    | S   | B     | 198 | 3    | S   | B     |
| 029 | 1    | S   | C     | 114 | 2    | S   | C     | 199 | 3    | S   | C     |
| 030 | 1    | S   | D     | 115 | 2    | S   | D     | 200 | 3    | S   | D     |



Table 1, (continued)

| No. | Zone | U/S | Char. | No. | Zone | U/S | Char. | No. | Zone | U/S | Char. |
|-----|------|-----|-------|-----|------|-----|-------|-----|------|-----|-------|
| 031 | 1    | S   | E     | 116 | 2    | S   | E     | 201 | 3    | S   | E     |
| 032 | 1    | S   | F     | 117 | 2    | S   | F     | 202 | 3    | S   | F     |
| 033 | 1    | S   | G     | 118 | 2    | S   | G     | 203 | 3    | S   | G     |
| 034 | 1    | S   | H     | 119 | 2    | S   | H     | 204 | 3    | S   | H     |
| 035 | 1    | S   | I     | 120 | 2    | S   | I     | 205 | 3    | S   | I     |
| 036 | 1    | S   | J     | 121 | 2    | S   | J     | 206 | 3    | S   | J     |
| 037 | 1    | S   | K     | 122 | 2    | S   | K     | 207 | 3    | S   | K     |
| 038 | 1    | S   | L     | 123 | 2    | S   | L     | 208 | 3    | S   | L     |
| 039 | 1    | S   | M     | 124 | 2    | S   | M     | 209 | 3    | S   | M     |
| 040 | 1    | S   | N     | 125 | 2    | S   | N     | 210 | 3    | S   | N     |
| 041 | 1    | S   | O     | 126 | 2    | S   | O     | 211 | 3    | S   | O     |
| 042 | 1    | S   | P     | 127 | 2    | S   | P     | 212 | 3    | S   | P     |
| 043 | 1    | S   | Q     | 128 | 2    | S   | Q     | 213 | 3    | S   | Q     |
| 044 | 1    | S   | R     | 129 | 2    | S   | R     | 214 | 3    | S   | R     |
| 045 | 1    | S   | S     | 130 | 2    | S   | S     | 215 | 3    | S   | S     |
| 046 | 1    | S   | T     | 131 | 2    | S   | T     | 216 | 3    | S   | T     |
| 047 | 1    | S   | U     | 132 | 2    | S   | U     | 217 | 3    | S   | U     |
| 048 | 1    | S   | V     | 133 | 2    | S   | V     | 218 | 3    | S   | V     |
| 049 | 1    | S   | W     | 134 | 2    | S   | W     | 219 | 3    | S   | W     |
| 050 | 1    | S   | X     | 135 | 2    | S   | X     | 220 | 3    | S   | X     |
| 051 | 1    | S   | Y     | 136 | 2    | S   | Y     | 221 | 3    | S   | Y     |
| 052 | 1    | S   | Z     | 137 | 2    | S   | Z     | 222 | 3    | S   | Z     |
| 053 | 1    | U   | 0     | 138 | 2    | U   | 0     | 223 | 3    | U   | 0     |
| 054 | 1    | U   | 1     | 139 | 2    | U   | 1     | 224 | 3    | U   | 1     |
| 055 | 1    | U   | 2     | 140 | 2    | U   | 2     | 225 | 3    | U   | 2     |
| 056 | 1    | U   | 3     | 141 | 2    | U   | 3     | 226 | 3    | U   | 3     |
| 057 | 1    | U   | 4     | 142 | 2    | U   | 4     | 227 | 3    | U   | 4     |
| 058 | 1    | U   | 5     | 143 | 2    | U   | 5     | 228 | 3    | U   | 5     |
| 059 | 1    | U   | 6     | 144 | 2    | U   | 6     | 229 | 3    | U   | 6     |
| 060 | 1    | U   | 7     | 145 | 2    | U   | 7     | 230 | 3    | U   | 7     |
| 061 | 1    | U   | 8     | 146 | 2    | U   | 8     | 231 | 3    | U   | 8     |
| 062 | 1    | U   | 9     | 147 | 2    | U   | 9     | 232 | 3    | U   | 9     |
| 063 | 1    | S   | 1     | 148 | 2    | S   | 0     | 233 | 3    | S   | 0     |
| 064 | 1    | S   | 2     | 149 | 2    | S   | 1     | 234 | 3    | S   | 1     |
| 065 | 1    | S   | 3     | 150 | 2    | S   | 2     | 235 | 3    | S   | 2     |
| 066 | 1    | S   | 4     | 151 | 2    | S   | 3     | 236 | 3    | S   | 3     |
| 067 | 1    | U   | .     | 152 | 2    | S   | 4     | 237 | 3    | S   | 4     |
| 068 | 1    | S   | .     | 153 | 2    | S   | 5     | 238 | 3    | S   | 5     |
| 069 | 1    | U   | <     | 154 | 2    | S   | 6     | 239 | 3    | S   | 6     |
| 070 | 1    | S   | <     | 155 | 2    | S   | 7     | 240 | 3    | S   | 7     |
| 071 | 1    | U   | &     | 156 | 2    | S   | 8     | 241 | 3    | S   | 8     |
| 072 | 1    | S   | &     | 157 | 2    | S   | 9     | 242 | 3    | S   | 9     |
| 073 | 1    | S   | \$    | 158 | 2    | U   | .     | 243 | 3    | U   | .     |
| 074 | 1    | U   | -     | 159 | 2    | U   | <     | 244 | 3    | S   | .     |
| 075 | 1    | S   | -     | 160 | 2    | S   | <     | 245 | 3    | U   | <     |
| 076 | 1    | U   | /     | 161 | 2    | S   | &     | 246 | 3    | S   | <     |
| 077 | 1    | U   | ,     | 162 | 2    | U   | \$    | 247 | 3    | U   | &     |
| 078 | 1    | S   | ,     | 163 | 2    | S   | \$    | 248 | 3    | U   | -     |
| 079 | 1    | U   | %     | 164 | 2    | U   | -     | 249 | 3    | U   | /     |
| 080 | 1    | S   | %     | 165 | 2    | U   | ,     | 250 | 3    | S   | /     |



Table 1 (continued)

| No. | Zone | U/S | Char. | No. | Zone | U/S | Char. | No. | Zone | U/S | Char. |
|-----|------|-----|-------|-----|------|-----|-------|-----|------|-----|-------|
| 081 | 1    | U   | >     | 166 | 2    | U   | %     | 251 | 3    | U   | ,     |
| 082 | 1    | S   | >     | 167 | 2    | U   | >     | 252 | 3    | S   | ,     |
| 083 | 1    | S   | ?     | 168 | 2    | S   | >     | 253 | 3    | U   | %     |
| 084 | 1    | U   | 12-0  | 169 | 2    | U   | 12-0  | 254 | 3    | U   | >     |
| 085 | 1    | S   | 12-0  | 170 | 2    | S   | 12-0  | 255 | 3    | S   | >     |

U indicates the Unshift Code (word separator, 0-5-8, 6D); S indicates the Shift Code (record mark, 0-2-8, 6E). The less than symbol, <, (12-4-8, 4C) and greater than symbol, >, (0-6-8, 6E) appear as shown.

The input deck for the LINGRID subroutine consists of twelve cards. Cols. 79-80 contain a two-digit code indicating the range of numbers shown in Table 1 to be included on that card. The width codes are punched in two-digit fields as indicated in Table 2. Two blanks must follow the last width code; the rest of the card may have identification punched in it. The leading zeros in each width code may either be punched or left blank. The width codes for each character are obtained from tables for each grid furnished us by the G.P.O. giving the actual character and the relative width.

Table 2

| Cols. 79-80 | Numbers From<br>Table 1 | Cols.<br>Punched |
|-------------|-------------------------|------------------|
| 1L          | 001-026                 | 1-52             |
| 1C          | 027-052                 | 1-52             |
| 1N          | 053-066                 | 1-28             |
| 1.          | 067-085                 | 1-38             |
| 2L          | 086-111                 | 1-52             |
| 2C          | 112-137                 | 1-52             |
| 2N          | 138-157                 | 1-40             |
| 2.          | 158-170                 | 1-26             |
| 3L          | 171-196                 | 1-52             |
| 3C          | 197-222                 | 1-52             |
| 3N          | 223-242                 | 1-40             |
| 3.          | 243-255                 | 1-26             |

Note that all letters are used in both shift and unshift mode. In Zone 1, only four numerals are used in the shift mode. The pattern for the BCD special characters is different in each of the three zones.

The subroutine requires three parameters:

1. The address of an 80-byte area into which a card can be read.
2. The address of a DCB specifying a card reader.
3. The address of a fullword in which the effective address of a 1020-byte grid table is returned. (Effective address = actual address minus 4.) The address returned from each call will be 1020 greater than that returned for an earlier call.

Each table to be assembled requires a separate call to the subroutine, at which time the twelve cards to be loaded must be in place in the card



reader. The subroutine will then read in the set of twelve cards, prepared as shown in Table 2. These may be arranged in any order. The width codes punched on the card are successively converted to binary and stored in the correct location in the table area. When all have been converted, a basic table containing the zone, shift, and character designations is combined with the width codes to make the final table. The address stored in the location specified by the third parameter is the address that should be in the fourth parameter location of the LINCOMP subroutine.

The following errors can occur with the results indicated:

1. An attempt to store a fifth grid. The following messages will appear on the console typewriter: "LINGRID SR: Attempt to start fifth grid" and "EODAD entered." and the program will enter the EODAD routine that has been specified in the DCB supplied as a parameter to the subroutine.
2. A card that does not have one of the twelve codes specified in Table 2 in cols. 79-80. The following message will appear on the console typewriter: "LINGRID SR: Invalid card code" and the program will read the next card. Card reading will continue until twelve cards with valid codes have been read.
3. After all twelve cards have been read, if any of the 255 words in the table has a zero width code, the following messages will appear on the console typewriter: "LINGRID SR: Zero width code in grid area" and "EODAD entered." and the program will enter the specified EODAD routine. This error will usually be caused by two cards with identical identification codes, missing width codes, zero width codes punched, or a similar error.

This subroutine requires 5632 bytes of storage. The object deck has the identification LGRD.

The use of this subroutine should be routine. Sets of the necessary twelve cards have been punched for each of the grids that will be used. The programmer will need only to obtain the necessary cards and to call LINGRID subroutine for each grid that he plans to use on his job.

#### LINACC Subroutine.

This subroutine will allow the programmer to compile an access table which will provide a connection between the characters obtained from copy, characters added in editing, characters used to select plus or minus signs, etc. and the characters on the Linotron grid. The subroutine will prepare a 256-byte translate table arranged so that each access character will be assigned a binary number from 1 to 255. These numbers are used to indicate the numbers from Table 1, and they appear on the type specimens, pages 41 and 43. This binary number, multiplied by four, will be an increment to select zone, shift, character, and width representations from a grid table stored by the LINGRID subroutine.



Cards to be used with this subroutine should be punched in the following format:

Col. 1. The character # (3-8) to indicate that each character on the card is punched as is, or the character @ (4-8) to indicate that hexadecimal notation has been used to indicate each character on the card.

Cols. 2-79 may be used to punch as many five- or six-character fields as may be desired.

For "character" notation each field should be punched C=nnn

For "hexadecimal" notation each field should be punched XX=nnn

where C indicates any punched character, XX indicates the two-character hexadecimal notation (0-9 ABCDEF), and n indicates one of the ten numerals 0-9.

A comma (0-3-8) must be used to separate each five- or six-character field. It must be omitted after the last field on the card. All leading zeros in the numerical fields must be punched (e.g. position 7 must be punched 007; 13 must be punched 013). The = sign is merely used as spacing; it may be left blank or contain any character.

The last card of a set must have a # (3-8) in column 80.

This subroutine requires three parameters:

1. The address of an 80-byte area into which a card can be read.
2. The address of a DCB specifying a card reader.
3. The address of a fullword to which the address of a 256-byte access table will be returned. Each address returned will be equal to that from a previous call +256.

Each access table to be assembled requires a separate call to the subroutine, at which time the access table cards should be in place in the card reader. Each card is then read and its contents processed. When the "end-of-set" card is read the subroutine then returns the address of that grid table to the calling program.

Room is reserved for eight access tables. Usually there will be one access table for each grid table. However, on some jobs it may be desired that copy letters and numerals should specify ordinary type for a part of the job, while on another part these same characters should select bold-face type. The address returned by the LINACC subroutine should be used as the third parameter in the LINCOMP subroutine.

Two examples are given. In the first the ten numerals and the characters L, H, A, and . are to select the corresponding Spartan Heavy characters from G.P.O. Grid No. 3. These characters are all in Zone 1.

#0=053,1=054,2=055,3=056,4=057,5=058,6=059,7=060,8=061,9=062

#L=038,H=034,A=027,.=067



In the second example the binary numbers 1 through 10 are used to select the numerals from Zone 3, unshift.

@01=224,02=225,03=226,04=227,05=228,06=229,07=230,08=231,09=232,0A=223

Access characters need not be specified for all 255 characters on the grid. Those unspecified will remain binary zero.

Any number of access characters may refer to the same character on the grid.

The following errors have the results indicated:

1. An attempt to store more than eight access tables (i.e. more than eight calls to the subroutine). An error message "LINACC SR: Tried to store ninth table. EODAD entered." will be printed on the console typewriter and the EODAD routine specified in the DCB furnished the subroutine will be entered.
2. Duplicate access characters. The binary number associated with the last one processed will be the one stored in the table; no error will be indicated.
3. Invalid character in hexadecimal notation (character other than 0-9, ABCDEF). The invalid character will be replaced by a binary zero half-byte; no error will be indicated.
4. Invalid numerical field (i.e. one which lacks three consecutive numerical characters). The field will be skipped. If this error is due to a short field, it is probable that the comma will be missing from its place further on on the card and the rest of the card will not be processed. No error will be indicated.
5. Number greater than 255. The incorrect number will be stored modulo 256; no error will be indicated.
6. Neither # or @ in column 1. The card will be checked for # in column 80 and then the next card will be read. No error will be indicated.

This subroutine requires 2672 bytes of storage. The identification on the object deck is LACC.

#### LINHEADS Subroutine.

The purpose of this subroutine is to provide a storage area for data coded directly in Linotron control characters and grid designations and to return addresses and length designations in a form that can be used by the LINWRITE subroutine. For every job there will be a number of items such as headings, rules, etc. which are the same on each frame. These can be coded into Linotron format, punched, and stored using the LINHEADS subroutine.



When a particular section is needed, its designation can be selected and given as a parameter to the LINWRITE subroutine.

Data to be introduced to the LINHEADS subroutine should be punched as described below. All Linotron data that are to be assembled in one block and called into place at one time on the frame should be designated as a set. (There is, of course, no requirement that such data be contiguous on the frame, and a single set could conceivably designate several headings located at several places on the frame, designation for rules, and include control characters for grid changes, turnpage, etc.)

Cols. 1-2. These must contain any two non-blank characters. All cards belonging to one set must contain the same two characters, and be read in order. Whenever a change of characters is recognized in this field, this will be used to indicate the end of a set.

Cols. 2-80. These columns may be used to code the data to be stored by the subroutine. A group mark (12-7-8,4F) is used as a signal to indicate end of data. Comments or identification may be punched as desired following the group mark.

The Linotron character for delta-x (11-0) must be punched using multiple punching. Since this character will be used more frequently than any other control code, it has been arranged that the ! (11-8-2,5A) will translate into 11-0, the delta-x symbol. Similarly / (12-8-2,4A) will translate into 12-0, one of the Linotron character codes. The translate table is so arranged that all other characters translate into themselves, except that characters not in the Linotron repertoire will translate into blanks.

An area of 10000 bytes has been reserved to store data. Any number of sets may be stored if they will fit in this space. The last set must be followed by a blank card (i.e. blank in col. 1) which will cause the subroutine to return to the calling program.

Three parameters will be required:

1. The address of an 80-byte area into which a card can be read.
2. The address of a DCB specifying a card reader.
3. The address of a storage area, aligned on a fullword boundary, to which data for each set will be returned. Two fullwords (8 bytes) must be included for each set. The following will be returned for each set: Word 1 is address of storage area in which the set is stored. Word 2 contains a halfword indicating number of bytes in the set and a halfword containing the two-byte identification for the set. This format is that required for parameter 2 of the LINWRITE subroutine.

Usually the programmer will know in what order his sets have been stored and the address of any particular set will be some multiple of eight displaced from the address used in parameter 3. Sometimes a "table lookup" scheme could be used, using the two-byte designation as the field to be matched. In either case the address wanted will be in one of the general registers where it can be stored as parameter 2 for the LINWRITE subroutine.



Ordinarily only one call to the LINHEADS subroutine will take place in a program. The effect of a second call will be to replace the data already stored, starting at the beginning of the LINHEADS storage area. If the new data are shorter than that first stored, then sets at the end of the string may still have valid designations from the earlier call.

In some applications it may be useful to arrange data on separate cards so as to be able to reproduce most of the data from the first card onto those that follow.

The following errors can occur with the results indicated:

1. An attempt to exceed the 10000 bytes allocated for storage. The console typewriter message "LINHEADS SR: Compilation space gone; EODAD entered." will appear and the EODAD routine given in the DCB furnished the subroutine will be executed.
2. Data card with no group mark on it. The console typewriter message "LINHEADS SR: Data card has no terminal group mark." will appear. The card will be skipped and the program will attempt to process the next card.
3. Insufficient space left in the calling program to store two fullwords per set. The subroutine will store as many addresses as required, erasing what follows. The result of this action is unpredictable.
4. Address in parameter 3 is not aligned on a full word boundary. A program interruption (Specification, 06) will occur.
5. No blank card to end set. The subroutine will read cards until all are gone or space is exhausted.

This subroutine uses 10894 bytes of storage. Its identification on the object deck is LNHD.

#### LINCOMP Subroutine.

This subroutine is used to provide a convenient way to convert data as they are likely to exist on the input medium into the corresponding grid characters using the tables assembled by the LINACC and LINGRID subroutines, and to add many of the necessary Linotron control characters. Usually it will be necessary to "edit" the input data for such things as adding punctuation, converting overpunched signs into characters that can be used to select plus or minus signs, provide for zero suppression, modify numerals to select bold face or italic numerals, etc. The composition using this subroutine works on a line basis; each line (i.e. a string of characters all of which have the same y-coordinate) will need a separate call to the subroutine.

Five parameters will be required:

1. The address of two consecutive halfwords: The value of  $y_0$  in the



first halfword and of  $x_0$  in the second halfword.

2. The address of an assembly area into which the edited character string to be processed by the subroutine is found.
3. The address of the address of an access table. This address may be returned from the LINACC subroutine.
4. The address of the effective (actual address minus 4) address of a grid table. This address may be returned from the LINGRID subroutine.
5. The address of two fullwords which will be returned in the following form: Fullword with the address of the line just composed, halfword with the number of assembled characters, halfword with the delta-x value of the compiled line. The first word and a half are in the format required by the LINWRITE subroutine; the last halfword may be used to compute centering, quadding, or in checking against precomputed values.

Each compiled line will start with a y-origin code, followed by the halfword value of  $y_0$  unpacked into five BCD numerals; then an x-origin code, followed by the halfword value of  $x_0$  unpacked into five BCD numerals. The characters stored in the assembly area will then be processed. It will be necessary to start each line with a Point Size Recognition code. The first character in the string will therefore be considered to be the character which will select a Point Size Recognition code thereafter in the string.

A number of characters have been given a special function that will be described. All of the other characters may be used as access characters used to select one of the 255 characters from the grid. The subroutine starts by processing each character in turn. If the character tested is an access character, then it is translated and tested against the access table (parameter 3). If a zero is returned, this will be treated as an error. Otherwise the number from the access table is multiplied by four and used as an index to increment the effective address of the grid table (parameter 4). The zone and shift codes are each tested against those previously used; new codes are introduced only as needed. The Linotron character is then stored and the width code is multiplied by the type size and the product added to the total delta-x.

The following characters have been assigned special functions and may not be used as access characters:

1. The programmer-supplied code to specify Point Size Recognition (See above). It must be followed by one of the eight Linotron type size codes (5,6,7,8,0,@,=, or S). Failure to have one of these codes will be indicated as an error.
2. FF. (Hexadecimal notation will be used for the rest of the codes; these characters are not on the "standard keypunch character" list.) This code will be used to indicate the end of the character string.



3. 00. This code is a null character and does nothing. Its usefulness comes when assembling variable length data into a given space. The shorter fields can have their unused character space replaced with binary zeros.

4. FE. This code will be used to indicate that a delta-x code (11-0) is to be assembled followed by the five BCD numerals indicating the amount of the delta-x. The FE code should be followed by a halfword indicating the amount of the delta-x. The halfword must be aligned on a halfword boundary; therefore the FE character must be followed by a padding byte if it is not located on an odd byte. The value of delta-x will be added to the delta-x sum.

5. FD. This character will be replaced by the Print Off code ( , 12-5-8, 4D).

6. FC. This character will be replaced by the Print On code (\*, 11-4-8, 5C).

7. FB. This character will be used to indicate that a "literal" follows. All of the characters until a second FB will be moved into place as is. No check is made to see if these are valid Linotron characters nor is the delta-x count increased. The zone and shift code test locations are blanked so that the next character obtained from an access character will have new zone and shift codes. This "literal" feature will allow such coding as that for synthetic bold, to introduce new y- and x-origins, or used as an alternate method of introducing coding that could have been introduced by the normal operation of the subroutine.

This subroutine uses register 15 to indicate that an error-free return has occurred. If the return is normal, register 15 will contain a binary zero. Other errors will cause the binary numbers 4, 8, or 12 to be placed in the register. These numbers could be used to select error correction routines or the following sequence of instructions could be used following the return from the subroutine:

```
LTR 15,15      ERROR is the address of an error routine or to
BNZ ERROR.     the sequence of instructions to end the job.
```

The errors that may occur include:

1. The subroutine allows only 1024 bytes into which to assemble a line of Linotron coding. If this space is exceeded then the console typewriter message "LINCOMP SR: Compilation space exhausted. RC=4" will be printed. Register 15 will contain a 4.

2. If the character indicated as a Point Size Recognition code is not followed by one of the eight valid Linotron size codes, the console typewriter message "LINCOMP SR: Invalid or missing typesize code. RC=8" will be printed. Register 15 will contain an 8.



3. If the character used as an access character does not reference a non-zero character in the access table used, then the console typewriter message "LINCOMP SR: Copy character not in access table. RC=12" will be printed. Register 15 will contain a 12. If this error should occur it will probably be caused by the cards for assembling the access table being incorrectly coded, a card missing, or a similar error.

4. If the address for parameter 5 is not on a fullword boundary then a program interruption (Specification C6) will occur.

This subroutine uses 2008 bytes. Its identification on the object deck is LCMP.

As an illustration of some of the ways that one can use the LINCOMP subroutine a hypothetical star catalog is used as an example.

The form of each line is to be

1234   11 14 32.123   +16 16 16.16   54.45

where the star number, right ascension, declination, and epoch are the four fields shown. Leading figures which are underlined are to be made blank if they represent leading zeros (i.e. zero suppression). In addition, the hours of right ascension are to be printed only for the first, sixth, eleventh . . . lines, or if the hour value differs from that on the preceding line. The input medium contains a 26-character record for each star in the following form

12341111432123+161616165445.

All data are numerical except the plus and minus signs which are denoted by 12 or 11 punches. The type size will be eight point. Numerals are set 9 relative or 72 absolute units. Plus and minus signs are 12 relative units = 96 units. The decimal point is 4 relative units = 32 units. A space of 6 relative units will be left between hours, minutes, and seconds or 48 absolute units. The space between fields will be  $1\frac{1}{2}$  ems or 27 relative units = 216 absolute units. Each of the numbers and the 11 and 12 punches can be used in the access table to select the proper numbers from the grid. It is necessary to edit in a symbol to access a period; a period will be as suitable as anything. It is necessary to specify delta-x values between the fields and to separate the hours or degrees, minutes, and seconds.

The character T is used as the Point Size Recognition symbol so that the 26 copy characters can be edited into the following pattern:

T 8          FEx (216)    FEx (48)    FEx (48)    .        FEx (216)   

FD FC FE(48)    FEx (48)    .        FE(216)    .        FF

Here the underlines indicate the "holes" into which the copy characters are to be placed. The numerals in the parentheses represent halfword (binary)



numerals indicating the delta-x value to be used. When the character FE is followed by an "x", this indicates the padding necessary to force half-word alignment. This pattern uses 60 bytes.

After the copy characters have been inserted, the zero suppression routines can take place. A routine is used to compute the  $y_0$ -coordinate and to place it in the first parameter location. The zero suppression routine for the star number should replace all leading zeros by binary zeros in the editing pattern and add 72 units to the starting  $x_0$  value for each zero suppressed. This  $x_0$  value is then stored in the second halfword of parameter 1. Zero suppression for the right ascension can be obtained by storing a 288 in the halfword location following the first FE if the first digit of the right ascension is zero, and replacing the zero by a binary zero. This same scheme will be repeated for the second digit of the hours which are to be printed only on the first line of each group of five or when the hour has changed from the previous line. The first digit of the degrees of declination is to be suppressed if it is a zero. Here another scheme has been used. The digit location has been preceded by a Print Off symbol and followed by a Print On symbol. If the digit should be a zero, these are left in place; otherwise they should be replaced by binary zeros, in which case the digit will print.

It has already been remarked that parameters 3 and 4 of the LINCOMP subroutine are to contain addresses that are returned by the LINACC and LINGRID subroutines. Ordinarily this will involve a considerable amount of transferring addresses to the address lists for the Calls to the LINCOMP subroutine. It is suggested that the actual Call Macros for this subroutine be used sparingly. This can be done by branching and linking to a single location which contains a Call to LINCOMP, which is followed by a branch to the register used to branch and link.

A more complicated illustration is the approach used in the program to compose H. O. 229. Two grids were used: Grid 1, the U. S. Naval Observatory Grid; Grid 2, G.P.O. Grid 3, containing the Spartan Heavy type. Two Calls were made to LINGRID to store the effective addresses of the grids in LIST1+12 and LIST2+12, respectively. The addresses of the access tables assembled using the LINACC subroutine were similarly stored in LIST1+8 and LIST2+8. These two fields had the format

```
LIST1    DC    A(YX,ASSEMBLY,0,0,COMPOUT)
LIST2    DC    A(YX,ASSEMBLY,0,0,COMPOUT)
```

Here YX was a fullword into which were placed the  $y_0$  and  $x_0$  coordinates for each line before the subroutine was entered; ASSEMBLY was a storage area large enough to hold the pattern that would be required for the longest line, and COMPOUT was the address of two fullwords to which LINCOMP would return its address, character count, and delta-x value. No Call Macro was used. Instead, register 1 was loaded with the address of LIST1 when the Naval Observatory grid was wanted; LIST2 when the Spartan Heavy Grid was wanted. Then a Branch and Link instruction using register 10 was made to CALLC where the following sequence followed:



```

CALLC  L   15,=V(LINCOMP)
        BALR 14,15
        LTR  15,15          Test for error return
        BNZ  $              $ was location of the closedown routine;
                           under the assumption that any error
                           found would require reprogramming

        LA   9,COMPOUT
CALLW   CALL LINWRITE,(OPBYTE,(9))
        BR   10

```

Each use of the LINCOMP subroutine also requires a Call to LINWRITE so it is convenient to have the two subroutines arranged as shown.

The entry point CALLW could also be used to transfer a character string from the LINHEADS storage area. The sequence

```

LA      9,HEADLIST+(xx)*8
BAL     10,CALLW

```

will perform the function desired. The location OPBYTE contains a binary zero; this can be changed to a binary one for the last use on each frame. This approach requires that the assembly area remains constant, new data being moved in each time the subroutine was used. It would be possible to change the address in the address list if this should be advantageous.

### Linotron Tape Print Program.

A Linotron Tape Print Program has been written. This program is useful for checking tapes before sending them to the G.P.O. and for general debugging purposes. Note that a general purpose tape print program (such as DEBE), will not usually be too satisfactory since the symbols for shift, unshift,  $x_0$ , and delta-y all print as blanks and many of the other common control symbols will print alike.

The tape to be printed must have been written in odd-parity, seven track tape, 556 cpi., and will of necessity be used on tape unit 180.

The user will have to punch one control card for each run. This is read as data under program control (not to be confused with the system control cards), which will specify which frames are to be printed or skipped. All of the records which refer to a particular frame will be printed as one group. The format of this card is as follows:

Each field is 11 digits long: 1-11, 12-22, 23-33, . . .

A maximum of seven fields can be used.

Each field has the format

|      |        |                                                    |
|------|--------|----------------------------------------------------|
| 1    | P or S | P indicates Print; S indicates Skip.               |
| 2-6  | =nnnn  | nnnn is number of the first frame in the group.    |
| 7-11 | =mmmm  | mmmm is the number of the last frame in the group. |

Frame numbers refer to the actual frame numbers written on the tape (positions 9-13 on the first record of each frame) and all leading zeros must be punched. The = sign (6-8) is the Frame Recognition Symbol.



The card should be punched to alternately skip or print groups. The first and last frame in the group may be the same (i.e. a single frame group). It is necessary that the numbers from field to field form a sequence with no gaps. For example, if one had a tape with 40 frames on it and it is desired to print the first two, frame 19, and frame 38-40, the following should be punched on the control card:

P=0001=0002S=0003=0018P=0019=0019S=0020=0037P=0038=0041.

No frame numbers may be omitted; the second field written as S=0004=0018 would be indicated as an error. Its effect would be to cause the message "FRAME NUMBERS OFF IN EITHER TAPE OR LEAD CARD" and a typewriter printout of the first 32-bytes of the record to be printed on the console typewriter, after which I/O will be closed. Giving a number larger than the last frame on the tape has the effect of causing an end of data set condition, causing the tape to rewind and unload and the job to end.

The program starts by spacing over the fanfold and printing a heading. The tape is checked to see if the first two bytes of the first record contain a Precedence B and Job Recognition Code. If this fails, the message "FIRST RECORD DOESN'T START A FRAME" and a printout of the first 32 bytes from the tape will be printed on the console typewriter and the job will be closed.

For each record printed, a heading containing the record number (counted consecutively from the beginning of the tape, including any skipped frames), the job number, and the frame number taken from the tape at the beginning of each frame is printed.

Each 2048 character record is printed in 16 128-character lines. Each character on the tape is translated into a two-character representation, printed on top of each other. There are enough blank second characters to give the effect of double spacing. Each letter or numeral that is used to represent a grid position will be translated into itself over a blank. The eleven BCD special characters that are used to represent a grid position will be printed as shown in Table 3.

Table 3

| EBCDIC | Punch  | Hex | Regular<br>Printing | Print | EBCDIC | Punch | Hex | Regular<br>Printing | Print |
|--------|--------|-----|---------------------|-------|--------|-------|-----|---------------------|-------|
| .      | 12-3-8 | 4B  | .                   | .     | ,      | 0-3-8 | 6B  | ,                   | ,     |
| <      | 12-4-8 | 4C  | )                   | )     | %      | 0-4-8 | 6C  | (                   | (     |
| &      | 12     | 50  | &                   | &     | >      | 0-6-8 | 6E  | ,                   | ,     |
| \$     | 11-3-8 | 5B  | \$                  | \$    | ?      | 0-7-8 | 6F  |                     | +     |
| -      | 11     | 60  | -                   | -     |        | 12-0  | C0  | &                   | +     |
| /      | 0-1    | 61  | /                   | /     |        |       |     |                     | 0     |



The Precedence A and the Precedence B codes will be translated  $\bar{A}$  and  $\bar{B}$  respectively. The other special characters which are used as Linotron control characters will be translated into two-character combinations which often will be an abbreviation of the function to be performed. Note that the same character will translate into two different codes according to what precedence code is in effect at the time. These are indicated in Table 4.

Table 4

| Precedence A |                   |       | Precedence B |                     |       |
|--------------|-------------------|-------|--------------|---------------------|-------|
| Hex          | Function          | Print | Hex          | Function            | Print |
| E0           | Shift             | S     | E0           | Shift               | S     |
|              |                   | H     |              |                     | H     |
| 6D           | Unshift           | U     | 6D           | Unshift             | U     |
|              |                   | S     |              |                     | S     |
| 7F           | X-Origin          | X     | 7F           | Aspect Ratio Recog. | A     |
|              |                   | X     |              |                     | R     |
| 7E           | Y-Origin          | Y     | 7E           | Frame Recognition   | F     |
|              |                   | Y     |              |                     | M     |
| 7C           | Point Size Recog. | S     | 7C           | Job Recognition     | J     |
|              |                   | +     |              |                     | B     |
| 4D           | Non-Print         | N     | 4D           | Special Face Recog. | S     |
|              |                   | P     |              |                     | *     |
| 5E           | Delta-Y           | D     | 5E           | Page Orient Recog.  | P     |
|              |                   | Y     |              |                     | O     |
| DO           | Delta-X           | D     | DO           | Stop                | \$    |
|              |                   | X     |              |                     | \$    |
| 5C           | Print On          | P     | 5C           | Film Advance        | F     |
|              |                   | R     |              |                     | A     |
| 7B           | Zone Recognition  | Z     | 7B           | Grid Recognition    | G     |
|              |                   | N     |              |                     | R     |
| 5D           | Rules Recognition | R     |              |                     |       |
|              |                   | =     |              |                     |       |
|              |                   |       | C6           | Focus               | F     |
|              |                   |       |              |                     | *     |
|              |                   |       | 6B           | Overlay             | O     |
|              |                   |       |              |                     | V     |

The Point Size Recognition code will cause the character which follows to be converted into two numerical digits to indicate the type size in decimal notation (e.g. 0 1 1). All of the other characters which follow a recognition code 5 0 4 are BCD numerals and these will print satisfactorily as is. Blanks on the tape will print as blanks; characters not in the Linotron repertoire will print as double asterisks.

An object deck for this program has been assembled with all of the necessary control cards. The jobname is LINTAPE; the DD cards for the tape, printer, and card reader have the DD names LINTAPE, OUTPUT, and INPUT, respectively. The loading procedure is normal; the tape should be mounted on Unit 180.



### Job Planning.

This section will not be a complete explanation of how to do a Linotron job but will indicate some points that are apt to be overlooked.

The first step for any job should be the design of a typical page. If the job is one of reproducing an existing publication this is usually easier since the arrangement of things to be printed, column arrangement, size of type, and so forth are generally fixed. Often there will be the question of improvements and minor changes and these should be investigated.

When a new job is to be printed the problem is generally more difficult. An attempt should be made to see how similar tables have been printed in our publications and in those of other organizations. There will be more problems and minor details to be taken care of than will be apparent at first. An attempt should be made to anticipate as many of these as possible rather than have them show up in the middle of coding the actual instructions. A careful layout on ruled paper is usually desirable.

Some of the things that will have to be decided include:

1. The width and depth of the page. These must fit the page size of the publication which is generally fixed. Sometimes the Linotron frame size (8 x 10½ inches) will be the determining factor. The width will usually depend on the number of columns to be printed. Sometimes the problem will be to squeeze everything so that it will fit a fixed width, or the other problem when it is necessary to set everything quite widely spaced in order that the page will not appear to be too narrow. The depth of the page will usually depend on the number of rows of tabular material that it is desired to set on each page and the size of type used. The change of point size by one point will usually mean that an additional five to ten lines can be printed on each page, and if the tabular material is extensive, this may mean a decrease of several pages for the job. As an example, when planning the printing of the rectangular coordinates of the minor planets using Linofilm (Astronomical Papers of the American Ephemeris, Vol. XX, Pt. II), the size of type was changed from ten to nine point and an additional ten lines were added to each page, making the book sixteen pages shorter than the previous edition (Vol. XVI, Pt. III).
2. Running heads and page numbers. It will often be necessary to continue the pagination and the system of running heads from that used for the text. Sometimes odd and even pages will have different running heads. Note that page numbers usually appear flush right on the odd pages, flush left on the even pages. This may be a moderately intricate programming problem. Often the program will require a page number counter and provision will have to be made for resetting it when starting up after an interrupted run.
3. Headings. These usually appear in larger type and often in a different style of type. The type faces available on the grids



usually go quite well together. Tabular material set in both Spartan Book or Bell Gothic will look well with Spartan Book, Spartan Heavy, or Trade Gothic headings. The spacing between lines is quite important for good appearance. A careful inspection of the first set of proofs is valuable to determine where the spacing can be improved.

4. Tabular material in columns. For most jobs this will be set in either 8- or 10-point. As explained above, the size will have an important effect on the number of lines that can be printed on a page. It is not necessary to allow exactly the same space for each line as the type size. If more space is allowed, this is called "leading" (rhymes with spreading) and additional white space will appear between each line. There is a good deal of white space with each character anyway, and tabular material set 8-point on 8-point body will appear quite open. Sometimes negative leading is used (e.g. the Air Almanac on Linofilm is set 9-point on an 8-point line). When space is left after a group of lines the space between each group is usually slightly less than half of the type size being used.

The horizontal spacing between fields should also be worked out carefully. If we assume that numerals are 9 relative units wide, the spaces between such things as degrees and minutes can usually be made about 6 to 8 relative units with 9 to 15 relative units between other main fields. Zero suppression often makes the spacing between columns appear wider and its effect should be allowed for in determining the spacing required between columns.

When differences are printed, it will usually be necessary to have these in smaller type on the same line (even differences) or on an offset line for odd differences or for critical tables. Both cases will usually require the lines to be set separately even in the case where two type sizes are to be set on the same line, since the y-coordinate refers to the z-line and the space between this and the actual base of the figures will be different for two different sizes of type.

It will be a good idea to plan each job so that it will not be necessary to do it all in one run, and to include a provision for terminating a run in such a way that the necessary "close-down" procedures are followed on the output tape. Procedures should be developed to allow the output tape to be spaced to the end of previously completed work, and for input files to be spaced to their starting points. While it will often turn out that the production run will be made in one pass, it will often happen that a checkout run will be done on only a few sample pages, and that these features will be of value then.

H.O. Pub. No. 229.

H.O. 229 is a six-volume navigational table which was one of the Naval Observatory's first efforts at large-scale electronic type setting using Linofilm. Each volume consists of computed altitudes and azimuths for one



degree intervals of declination, hour angle, and latitude. In each volume the declinations run from  $0^{\circ}$  to  $90^{\circ}$ ; the hour angles from  $0^{\circ}$  to  $360^{\circ}$  (arranged by quadrants); and the latitudes cover a  $15^{\circ}$  range in each volume. Each volume has two types of pages, one for latitude the same name (sign) as declination; the other for latitude contrary name to declination. The contrary pages also contain a section of "same" entries for hour angles in the second and third quadrants.

The computed tabular material for all six volumes has been computed and written on 12 tapes. Similarly, 12 tapes were made with Linofilm coding, and three volumes (Vols. 6, 5, and 4) have been processed although only one has had all of the necessary work needed for the complete job.

The Linofilm process has the following disadvantages for this job:

1. It is not possible to compose either the rules or the minor headings on each page. This information has been added using overlays.
2. On the "contrary" pages it is necessary to have "stairstep" rules drawn by hand to separate the section (L.H.A. less than  $90^{\circ}$ ) from that portion of the page assigned to the "same" portion (L.H.A. greater than  $90^{\circ}$ ).
3. The Linofilm machine spaces the film after each line. The error in this process is quite small but on a page as long as this it may accumulate to as much as a pica, and varying somewhat from page to page. This makes it difficult to fit the overlays.
4. The conversion from magnetic tape to paper tape is not perfect. Bits dropped or added cause annoying sporadic errors.

The Linotron process will allow the complete page to be set with rules and "stairsteps". Both kinds of pages will have to be enlarged 28% because of the size limitations for both processes.

The decision to start our Linotron processing for this job was made for these reasons, besides the fact that this would be a very good job to use in order to gain experience with the Linotron system. Much of the detailed work concerning column widths, type sizes, spacing between lines, and so forth could be taken without change from the Linofilm programming.

The rest of this section will explain in some detail the job organizing and the main lines of the Linotron program that was written.

The first consideration was to decide on what form the input tape should take. The original input tapes used with the Linofilm program had several deficiencies. One of these was that a set of differences for the  $90^{\circ}$  declination line had not been computed. Another was that it had been planned to "tag" the differences in each case where the double second differences were greater than a specified limit, so that these differences could be printed in italics. These deficiencies had been taken into account in the Linofilm



program and nearly a third of that program was to correct them. Other things that the Linofilm program did was to compute the page number from a formula involving the hour angle and to compute the sixteen latitude headings from the initial latitude value on a header record.

Since many of these computational steps were rather complicated and the program steps were available on Autocoder cards, it was decided to use the 1410 program as a basis for a program which would do as many of the necessary computational steps as possible and to make a corrected tape rather than just to make a 7- to 9-track tape conversion. The original tape had each page on three records; the first and third gave the hour angle values printed at the top and bottom of the pages and indications whether it was a "same" or a "contrary" page. The tabular material was set as 1092 12-byte fields arranged in rows (8736 bytes). The new tape combined the heading material, page numbers, etc. with the tabular material into a new 8800-byte record. Provision was also made to eliminate the trailer and header records and to combine six input tapes into a concatenated file containing all of the material for Volumes 1, 2, and 3.

The program written for the Linotron run does the following:

1. A card is read which contains the number of records needed to be spaced on the output tape and the Linotron startup constants. (The first field is blank on the initial run.) A Call is made to LINFRAMX and the startup constants are put in place. The starting frame number must be changed each time a job is continued. This card is followed by the cards for the two grid and access tables used in the program and which are stored in place by two Calls to the LINGRID and LINACC subroutines.
2. The coding for ten different constant portions of the job are read from cards using the LINHEADS subroutine.
3. A control card is read with pairs of 5-character fields which specify the starting and stopping records on the input tape. This allows for spacing the tape to a given point before starting. On the check runs these fields were arranged to select special pages to make a good sample illustrating the changes that take place as the hour angle changes. For the production run, this card has simply one set of starting and ending records, usually for a half volume.
4. The processing for a frame begins with a Call to LINFRAME. The grid in turret 2 (containing Spartan Heavy) is specified in the startup constants so that all of the material set in Spartan Heavy is composed first.
5. The heading "Latitude SAME Name as Declination" at the bottom of each frame, the 91 declination values, and the symbols Dec. Hc d Z used as column headings are constant on every page and are coded in two sections taken from the LINHEADS storage area. The latitude values are taken from the input tape and edited and composed using the LINCOMP subroutine. The line set for the top of the



page is converted to one for the bottom of the page by changing the BCD characters giving the  $y_0$  value and leaving the rest of the line the same.

6. The rest of the Spartan Heavy type depends on whether a "contrary" or "same" page is being set. On the "same" page the L.H.A. values are set flush with the left margins. It is necessary to check whether one or two digits are contained in the first hour angle; the supplemental angle always has three digits. Since the same L.H.A. appears at the top and bottom of the page it is necessary only to change the  $y_0$  value for the composed line. The heading "Latitude SAME Name as Declination" is placed at the top of the page. The "contrary" pages have different L.H.A. values at the top and bottom of the page and these are set flush right. This means that the  $x_0$ -coordinate for each line is different depending whether there are one, two, or three digits in the first hour angle. The scheme used is to compose each line and then to subtract the delta-x value returned from the LINCOMP subroutine from the total width of the page and to use this as the  $x_0$  value. The heading "Latitude CONTRARY Name To Declination" is taken from the LINHEADS storage area. This concludes the material to be set in Spartan Heavy type.

7. At this point a grid change to the Naval Observatory grid is required. The necessary precedence codes and the Grid Recognition symbols have been included as the first characters of the coded minor headings that are called from the LINHEADS storage area. These headings are different for the "same" and the "contrary" pages. A line containing the degree and minute column headings is also taken from the LINHEADS storage area.

8. The tabular material is the same on the "same" and "contrary" pages. The characters are set line by line, 96 characters from the input tape going into each line. The  $y_0$ -coordinate is computed for each line from the formula  $D/5 \times 531 + D \text{ (modulo } 5) \times 91 + 875$ . This provides for each line being set 6-point on  $5\frac{1}{2}$  points plus 2 points for the space between groups of five. The editing routine to prepare the tape data edits each column's entry as one of eight items in the line. The altitude is processed first. This is the same for all items except for a possible lead zero being suppressed. All zero suppression is done by having the zero appear between non-print and print symbols. If a tag is found over the first digit of the altitude this indicates that a "stairstep" rule should be drawn under this item in this column. The  $y_0$ -coordinate is found, a small constant is subtracted, and the  $y_0$  value is converted to zoned decimal and put in its appropriate place in a pattern which specifies Rule Recognition symbols,  $x_0$  values, and delta-x values for later composition.

A shift is made to 5-point type before processing the differences. The sign overpunch is converted to a symbol to access either a plus or minus sign. This is suppressed between non-print and print symbols except when the sign appears as the first item in a group of



five or differs from that appearing on the preceding or following line. No sign is printed when the difference is zero. The differences may have the first digit zero suppressed. Italic differences are tagged with an 11-overpunch in the unit position. This zone is moved over the remaining figures and the characters 11-0 through R are used as access characters to select the 10 italic numerals from the grid. If the characters are set in italic a small dot is set 6-units wide after shifting back to 6-point type. Otherwise this character between non-print and print codes provides 36 absolute units between the differences and the azimuth.

One or two leading zeros may require zero suppression in the azimuths; otherwise there is no special problem. A loop to perform these operations is performed for 91 lines. LINCOMP is Called after each line when eight columns have been edited.

9. The basic rule pattern for each page is selected from the LINHEADS storage area. If "stairsteps" have been composed while setting the tabular data, then these are moved to the output tape. The page number is then taken from the input tape and composed at the bottom of the page. It is necessary to modify the  $x_0$ -coordinate to provide centering depending on whether there are one, two, or three digits in the numeral. This last Call to LINWRITE has the option byte changed to a binary one to cause the last characters to be followed by a film advance.

10. The input tape is arranged so that there are two consecutive records with the same L.H.A. values. The first of these is for the first eight degrees of latitude; the second for the last eight degrees. Tests for the L.H.A. value equalling the ending value are not made until after the second record has been processed. After this the Interrupt switch is tested (to provide a stop procedure under operator control) and then the L.H.A. field is tested against the end of block field on the control card. If more records remain to be processed these are done. In all cases when an end-of-job is recognized, a Call to LINFRAME is made to produce a new (blank) frame to be used as a trailer, followed by a Call to LINWRITE with the closing option which provides the necessary film advance and stop codes and types out how many records there are on the output tape. The other I/O units are closed and the job ended.

This program requires less than 2000 bytes, since the input tape is read in the Locate mode in which the address of the buffer rather than the tape characters are returned to the calling program.







## LINOTRON FUNCTION CODE REPERTOIRE

| B | A | 8 | 4 | 2 | 1 | PUNCHED<br>CARD<br>CODE | (101111)<br>PRECEDENCE A | (010000)<br>PRECEDENCE B   |
|---|---|---|---|---|---|-------------------------|--------------------------|----------------------------|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 8 2                   | SHIFT                    | SHIFT                      |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 8 5                   | UNSHIFT                  | UNSHIFT                    |
| 0 | 0 | 1 | 1 | 1 | 0 | 8 6                     | ABSOLUTE Y               | FRAME RECOGNITION          |
| 0 | 0 | 1 | 1 | 1 | 1 | 8 7                     | ABSOLUTE X               | ASPECT RATIO RECOGNITION   |
| 0 | 0 | 1 | 1 | 0 | 0 | 8 4                     | POINT SIZE RECOGNITION   | JOB RECOGNITION            |
| 1 | 1 | 1 | 1 | 0 | 1 | 12 8 5                  | NON - PRINT              | SPECIAL FACE RECOGNITION   |
| 1 | 0 | 1 | 1 | 1 | 0 | 11 8 6                  | DELTA Y                  | PAGE ORIENT RECOGNITION    |
| 1 | 0 | 1 | 0 | 1 | 0 | 11 0                    | DELTA X                  | STOP MACHINE               |
| 1 | 0 | 1 | 1 | 0 | 0 | 11 8 4                  | PRINT RECOGNITION        | FILM ADVANCE               |
| 0 | 0 | 1 | 0 | 1 | 1 | 8 3                     | TYPE FACE RECOGNITION    | GRID RECOGNITION           |
| 1 | 1 | 0 | 1 | 1 | 0 | 12 6                    |                          | FOCUS                      |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 8 3                   |                          | FORMAT OVERLAY RECOGNITION |
| 1 | 0 | 1 | 1 | 0 | 1 | 11 8 5                  | RULE RECOGNITION         |                            |



| LINOTRON FUNCTION CODE REPERTOIRE |                            |                          |  |  |  |  |  |  |  |
|-----------------------------------|----------------------------|--------------------------|--|--|--|--|--|--|--|
| PRECEDENCE A<br>(10111)           | PRECEDENCE B<br>(01000)    | FUNCTION<br>CARD<br>CODE |  |  |  |  |  |  |  |
| SHIFT                             | SHIFT                      | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| UNSHIFT                           | UNSHIFT                    | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| FRAME RECOGNITION                 | FRAME RECOGNITION          | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| ASPECT RATIO RECOGNITION          | ASPECT RATIO RECOGNITION   | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| JOB RECOGNITION                   | JOB RECOGNITION            | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| SPECIAL FACE RECOGNITION          | SPECIAL FACE RECOGNITION   | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| PAGE BRIGHT RECOGNITION           | PAGE BRIGHT RECOGNITION    | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| STOP MACHINE                      | STOP MACHINE               | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| FILM ADVANCE                      | FILM ADVANCE               | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| GRID RECOGNITION                  | GRID RECOGNITION           | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| FOCUS                             | FOCUS                      | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
| FORMAT OVERLAY RECOGNITION        | FORMAT OVERLAY RECOGNITION | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
|                                   |                            | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |
|                                   |                            | 0 0 0 0 0 0 0 0 0 0      |  |  |  |  |  |  |  |



## CENTURY EXPANDED GRID 1

1L 001-026

### Zone 1

a b c d e f g h i j k l m n o p q r s t u v w x y z

1C 027-052

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1N 053-066

0 1 2 3 4 5 6 7 8 9 A B C D

1. 067-085

. ) ; : [ & — - - ] , ( \$ % ! ? \* ' ‘

**RULE Weight 1**

2L 086-111

Zone 2

a b c d e f g h i j k l m n o p q r s t u v w x y z

2C 112-137

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

2N 138-157

0 1 2 3 4 5 6 7 8 9 T K L M N O P. Q R S

2. 158-170

· ; : & - U - , \$ ! ? ' ‘

RULE Weight 2

3L 171-196

Zone 3

a b c d e f g h i j k l m n o p q r s t u v w x y z

3C 197-222

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

3N 223-242

0 1 2 3 4 5 6 7 8 9 Y E F G H I & V W X

3. 243-255

z ) ; : [ j ] / & ( \$ ! ?

RULE Weight 4

5 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890ABCDEFGHIJKLMNPOQRSTUVWXYZ

6 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890ABCDEFGHIJKLMN

7 POINTS: ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890ABCDEFGHIJKLMN

8 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890ABCDEFGHIJKLMNOPQRSTUVWX

10 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890ABCDEF

12 POINT ABCDEFGHIJKLMN O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 1 2 3 4

14 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz

18 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz

$123456$      $123456$      $123456$

123456    123456    123456







1C 027-052

1N 053-066

1. 067-085

**RULE Weight 1**

2L 086-111 Zone 2

2C 112-137

2N 138-157

2. 158-170

RULE Weight 2

3L 171-196 . Zone 3

3C 197-222

3N 223-242

3. 243-255

RULE Weight 4

5 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

6 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

7 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

8 POINT ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

10 POINT ABCDEFGHIJKLMN O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 1 2 3 4 5 6 7 8 9 0

12 POINT **ABCDEFGHIJKLMN**OPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890

12 POINT **ABCDEFGHIJKLMN**  
14 POINT **ABCDEFGHIJKLMN**

18 POINT **ABCDEFGHIJKLMN O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o**

18 POINT

123456      123456      123456

123456      123456      123456